CLAIMS

We claim:

1. A system for processing transport stream data, the system comprising:

a framer module having

an input node to receive the transport stream data,

a data output node to provide a framer data which is a representation of the transport stream data, and

a data enable output node to provide a signal to indicate a valid data on the data output node;

a first parser module having

a data input node coupled to the data output of the framer module to receive the framer data,

a data enable input node coupled to data the data enable output node of the framer module;

a data output node to provide a first parser data when the framer data is a first data type, wherein the first parser data is a representation of the framer data,

a first data enable output node to provide a signal to indicate a valid first parser data on the data output node of the first parser, and

a second data enable output node to provide a signal to indicate the framer data is of a second data type.

2. The system of claim 1 further comprising:

a second parser module having

a data input node coupled to the data output node of the framer module to receive the framer data,

an enable input node coupled to the second data enable output node of the first parser module,

a data output node to provide a second parser data when the signal associated with the second data enable output node indicates the framer data is of a second data type, wherein the second parser data is a representation of the framer data;

a data enable output node to provide a signal to indicate a valid second parser data on the data output node .

3. The system of claim 2 further comprising:

a first memory having

a first input node coupled to the data output node of the first parser module to receive the first parser data,

and a second input node coupled to the first data enable output node of the first parser module; and

a second memory having

a first input node coupled to the data output node of the second parser module to receive the second parser data,

and a second input node coupled to the data enable output node of the second parser module.

4. The system of claim 3, where the first memory is a system memory, and the second memory is a video memory.

5. The system of claim 4, wherein the first parser module further includes an index output node to indicate a predefined data type, the index output node coupled to the first memory to identify a specific location within the system memory.

6. The system of claim 2, wherein the second parser module is a hardware parser.

7. The system of claim 6, wherein the second parser module has a modular layout that is substantially mutually exclusive of a modular layout of the first parser module.

8. The system of claim 6, wherein the second data type is video data.

9. The system of claim 1, wherein the data output node includes a plurality of nodes.

10. The system of claim 9, wherein the data plurality of nodes includes eight or more nodes.

11. The system of claim 1 further comprising:

a storage location to store a register set, the storage location having a first input node coupled to a status output node of the first parser module to receive status information, and a second input node coupled to a status output node of the second parser module to receive status information.

12. The system of claim 1, wherein the first input node of the storage location and the second input node of the storage location are a common input node.

13. A system for processing transport stream data, the system comprising:

a framer having a modular layout, the framer comprising

an input node to receive the transport stream data,

a data output node to provide a framer data based upon the transport stream data, and

a data enable output node to provide a signal to indicate a valid data on the data output node;

a first parser having a modular layout separate from the modular layout of the framer, the first parser comprising:

a data input node coupled to the data output node of the framer to receive the framer data,

a first data enable output node to provide a signal to indicate a first type of framer data,

a second data enable output node to provide a signal to indicate a second type of framer data.

14. The system of claim 13 further comprising:

a second parser having

a data input node coupled to the data output node of the framer to receive the framer data,

an enable input node coupled to the second data enable output node of the first parser,

a data enable output node to provide a signal to indicate the second type of framer data is to be stored.

15. The system of claim 14 further comprising:

a memory controller having

a data input node coupled to the data output node of the framer to receive the data of the predefined size,

a first enable input node coupled to the data enable output node of the framer,

a second enable input coupled to the first data enable output node of the first parser,

a third enable input coupled to the data enable output node of the second parser, and

a data output to provide data to a memory.

16. A method of parsing a data packet, the method comprising the steps of:

providing a start indicator to a first parser, the start indicator indicating a first data block of the data packet, the data packet having a predetermined number of data blocks;

analyzing at the first parser at least a portion of the first N data blocks after the start of the data packet to determine a data type of a subsequent data block of the data packet, wherein the subsequent data block is after the first N data blocks;

enabling a second parser to receive the subsequent data block when the data type of the subsequent data block is a first data type; and

enabling a third parser to receive the subsequent data block when the data type of the subsequent data block is a second data type.

17. The method of claim 16 wherein the first parser is a hardware parser.

18. The method of claim 17, wherein the second parser is a hardware parser.

19. The method of claim 18, wherein the first and second hardware parsers are modular and substantially physically separate from each other.

20. The method of claim 14, wherein a data block is a byte of data.

21. The method of claim 20, wherein the predetermined number of data blocks is 188.

22. A system for storing packetized data, the system comprising:

a means for receiving a transmitted data packet;

a first parser means for analyzing a header of the data packet before a payload header is received; and

a second parser means physically separate from the first parser means for analyzing the payload header.

23. The system of claim 22, wherein the first parser further analyzes the header of the data packet before a second byte of payload header is received.

24. The system of claim 23, wherein the second parser further analyzes the payload header before a second byte of payload data is received.

25. The system of claim 22 further comprising:

a memory controller to store a first portion of payload data in video memory before storing a second portion of payload data in video memory, wherein the first portion of payload data immediately follows the payload header, and the second portion of payload data immediately follows the second portion of payload data.

26. The system of claim 25, wherein the memory controller stores the first and second portion of payload data based upon the payload header.

27. A system for storing a data stream, the system comprising:

a data processing system having

a system memory having data port, and

a video memory having a data port;

a data stream demultiplexer having

a first data port coupled to the data port of the video memory to buffer video data associated with the data stream in the video memory, and

a second data port coupled to the data port of the system memory to buffer non-video data associated with the data stream in the system memory.

28. The system of claim 27, wherein the data processing system is a general purpose computer.

29. The system of claim 27, wherein the video memory is associated with a video graphics adapter.

30. The system of claim 27, wherein the data port of the system memory is a PCI (Peripheral Communications Interface) type interface.

31. The system of claim 30, where in the data port of the video memory is an AGP (Accelerated Graphics Port) type interface.

32. The system of claim 30, where in the data port of the video memory is a PCI type interface.

33. A method of for processing transport stream data, the method comprising the steps of:

receiving a transport packet having a header and a payload, wherein the payload is associated with a primary elementary stream (PES) which can be associated with video or non-video data, wherein non-video data includes video data that is not being used;

determining if the PES is a program clock reference (PCR) PES, wherein a PCR PES is a PES that is predefined to carry a program clock reference (PCR) currently used by a decoding system; and

parsing a first set of data in the header of the transport packet using a hardware adaptation field parser when the PES is a non-video PCR PES; and parsing a second set of data in the header of the transport packet using a hardware adaptation field parser when the PES is a video PCR PES, wherein the second set includes more elements than the first set.

34. The method of claim 33, further comprising the steps of:

storing at least a portion of the PCR PES in a system memory location when the PCR PES is a non-video PES; and

storing at least a portion of the PCR PES in a video memory location when the PCR PES is a video PES.

35. A method of for processing transport stream data, the method comprising the steps of:

receiving a transport packet at a hardware transport packet parser, the transport packet having a header and a payload, wherein the payload is associated with a primary elementary stream (PES) which can be associated with video or non-video data, wherein non-video data includes video data that is not being used;

saving at least a portion of the transport packet payload in a register set when the PES is a video PES.

saving none the transport packet payload in the register set when the PES is a non-video PES.